

Decision Taking as a Service

Jan A. Bergstra

Section Theory of Computer Science, Informatics Institute,
Faculty of Science, University of Amsterdam,
Amsterdam, The Netherlands.*

Abstract

Decision taking can be performed as a service to other parties and it is amenable to outtasking rather than to outsourcing. Outtasking decision taking is compatible with selfsourcing of decision making activities carried out in preparation of decision taking. Decision taking as a service (DTaaS) is viewed as an instance of so-called decision casting. Preconditions for service casting are examined, and compliance of decision taking with these preconditions is confirmed. Potential advantages and disadvantages of using decision taking as a service are considered.

1 Introduction

This paper has been written with the objective to state and answer the question to what extent it is possible and meaningful to use and to offer decision taking (DT) as a service, hereafter abbreviated as DTaaS.

For the definition of decision taking that I will use, I refer to [6]. That definition is unusual in the sense that a decision is an action rather than a result of an action.

1.1 Service casting and service casting preconditions

DTaaS will be understood as a substitution instance of the context $[-]aaS$ where DT is placed in the “hole”. Thus DTaaS abbreviates $[DT]aaS$. For an activity or entity or process X, $XaaS = [X]aaS$ is the service casting of X. In particular DTaaS represents the service casting of DT, provided that DT satisfies the constraints that any X must comply with for $[X]aaS$ to make sense, that is to be well-defined. These constraints are called service casting preconditions. The task of this paper is twofold: (i) to determine a reasonable account of the service casting preconditions in general, and (ii) to argue why DT, and to what extent, complies with these service casting preconditions.

*Author’s email address: j.a.bergstra@uva.nl.

1.2 Parametrized roles

S abbreviates service and $[-]_{aa}S$ stands for the service casting operator $X \rightarrow [X]_{aa}S$. S can be considered a role and the service casting operator can be understood as an instance of the so-called generic role casting operator which takes role Y to the role casting operator $[-]_{aa}[Y]$.¹

Below I will make use of the role casting operator $[-]_{aa}[F]$ where F stands for feature. $X_{aa}F$ views X (or instances of X) as a feature of entities or processes that realize some form of X. Speaking of $X_{aa}F$ indicates that an entity realizing X may keep its identity even if X is stripped from it.

Role casting preconditions indicate for which roles Y, $[-]_{aa}[Y]$ is a valid role casting operator. I will avoid this level of generality and I will assume that “service” satisfies suitable role casting preconditions without making an attempt to analyze these in detail.

1.3 Decisions and units

In [31] the observation is made that decision making research mainly stands on two feet: description and prescription; a similar dichotomy underlies the survey [32]. The same may hold for work on decision taking. I will avoid both description and prescription, and instead focus on the structure of organizational processes. I will understand decision making and decision taking as structural mechanisms independent of existing practice. Decision taking constitutes an organizational control mechanism.

Following [10, 11] I will speak of units to indicate all forms of organizational entities. Units may be equipped with a hierarchical structure with subunits at various levels. Sourcing (see [9]) are descriptions of sourcing relations between units. Outsourcing and outtasking are examples of sourcing transformations.

Decision making (DM) can be split in two parts: decision preparation (DP) and decision taking. In symbols: “DM = DP + DT”. Because DT is usually connected with leading roles and a unit will not function without leaders, it is implausible that DT is outsourced together with the sources (leaders) responsible for it.

Some authors (e.g. see [42]) read decision taking as the reception of a decision outcome in the role of staff responsible for realizing the goals of the decision taker. Although this is an acceptable interpretation of decision taking I will deviate from it. I will not have a special term for the action or process of taking notice and subsequent comprehension of a decision outcome. Other authors, e.g. in [37] use decision taking as an equivalent of decision making, a use of the terminology that I am not following either.

In [3] decision making and decision taking are distinguished in a way that comes close to the interpretation of [6] by accepting the real time character and the situational dependency of decision taking, in contrast with decision making, but without assuming the convention that a decision is an action. Below I

¹ $[-]_{aa}S = (\text{let } Y = S \text{ in})[-]_{aa}[Y]$.

will make use of so-called outtasking of decision taking. Admittedly this is an uncommon phrase; it is related to delegation of decisions as found in [36] and in [22].

In [41] the outsourcing of self-government is analyzed as a philosophical theme. The analysis leads the author to formulating doubts concerning the mentioned notion. These doubts are grounded in both the human nature of the outsourcing agent and in the comprehensiveness of the range of decisions supposed to be outtasked. In the terminology that I prefer to use, outtasking would be preferable to outsourcing for the use made of in in [41], because the human agent, viewed as a unit, cannot possibly export any of its internal sources (given today's biotechnology at least) to an external unit.

1.4 Organization of the paper

DT is contrasted with voting and promise issuing in Section 2, then in Section 3 DT is viewed as a feature of organizational behavior. A comparison with computer program structuring is made, and it is described how an organization may proceed if it makes no use, or limited use, of the DT feature. In Section 4 the concept of service casting, that is casting an entity class or an activity class (concept) as a service, is examined in rigorous detail. Several examples of service casting are considered and service casting preconditions as well as service casting obstacles are collected. In Section 5 the particular case of service casting DT is approached by indicating how DT complies with the service casting preconditions and avoids a match with the service casting obstacles. In Section 5.3 possible rationales for making use of DTaaS are surveyed and a preliminary risk analysis for DTaaS are provided. In Section 6 the intended audience for a theory of DTaaS is specified and the intended yield for those in the intended audience of acquiring familiarity with our theory of DTaaS is formulated. Finally in Section 6 some conclusions and directions for further work are mentioned.

2 Voting and promise issuing

With $X \subseteq Y$ (X is a subclass of Y) it will be formalized that every instance of X is an instance of Y . $X-Y$ denotes the class of instances of X that are not instances of Y . With ACT denoting the class of actions and with CH denoting the class of choices (acts of choosing from a menu of alternatives) it follows that $DT \subseteq ACT$, $CH \subseteq ACT$, while DT and CH are not disjoint and neither one is a subclass of the other. If by default (but not necessarily always) instances of X are also instances of Y that relation between X and Y is denoted as $X \subseteq_d Y$. It follows that $X \subseteq Y$ implies $X \subseteq_d Y$. Individual voting (IV) is a subclass of choice: $IV \subseteq CH$. Social choice (SC) is an aggregate of which IV 's are a part (component) rather than an instance. Concurrent IV 's are collected in a group vote (GV) which itself is embedded (used as a phase in) social choice. Now, $CH \subseteq_d ACT-DT$, $GV \subseteq_d ACT-DT$, and $SC \subseteq_d ACT-DT$.

Promise issuing (PI) is a class related to DT . Below $c(X)$ will denote the

complement of X. $PI \subseteq_d c(IV \cup GV \cup SC)$. Further $PI \subseteq_d ACT-CH$, that is, most most acts of issuing a promise are not choices. Obligation creating promise issuings (OCPI) are a proper subclass of PI, so $OCPI \subseteq PI$. Instances of OCPI are not decisions however: $OCPI \subseteq ACT-DT$. It then follows that $DT \subseteq PI-OCPI$.

For a class of actions will consider XaaS plausible (abbreviated: $pl(XaaS)$) if there for most instance of X it is conceivable in technical terms that these are provided as a service. I will consider XaaS implausible if it is not the case that XaaS is plausible. Now I will use the (defeasible) rule that if $X \subseteq_d Y$ & $pl(YaaS) \Rightarrow pl(XaaS)$. This rule can be applied with $X=PI-OCPI$ and $Y=DT$, leading to the somewhat unexpected conclusion that $pl([PI-OCPI]aaS)$, for which an opportunistic style of marketing may serve as an example.

These matters are far from obvious (and may well be disputed), and for that reason I will now expand on the underlying arguments which led to these assertions in some detail.

2.1 Voting, social choice, and decision

For the purpose of this paper the boundaries of decision must be laid down with more precision than has been done in [6]. In [6] the distinction between decision and choice has been discussed extensively, with the effect that (i) an act of choice need not be a decision and, (ii) that a decision need not involve choice.

However, between choice and decision there is voting, as well as social choice. Related to decision is promising (promise issuing). I hold that voting, social choice, and promising by default (that is in most cases) do not constitute decisions. Clearly understanding the difference between promising, voting, social choice, and decision is necessary because neither voting, or social choice, nor promising can be cast as a service.

2.2 Voting and social choice

In order to distinguish voting from decision taking I will make the following assumptions about voting:

1. Casting a vote (by an individual voter) is an act of voting.
2. Each act of voting is an instance of choice.
3. The result of casting a vote (by an individual voter) is an individual voting outcome (IVO).²

²An IVO is often referred to as a vote but that terminology is confusing as a vote may also refer to an act of voting, which I prefer as its meaning. It may even refer to the fact that an agent has the right to vote. The common use of the term vote seems to vary between static and dynamic interpretations just as the term test in software engineering does (see [29]).

4. Individual voters act concurrently,³ while together performing a group voting process.
5. A group voting process terminates and subsequently the plurality of collected IVO's can be aggregated to a single AVO (aggregate voting outcome) by means of some aggregation method.⁴

The design, analysis, and selection of aggregation methods for votings belongs to the area of social choice theory. There exists a large volume of theory on voting and social choice for instance [34, 43], most of it not in need of notions like IVO and AVO. An AVO is frequently referred to as a decision, thus reflecting a point of view that I do not share.

6. The AVO is merely the outcome of the overall voting process, and it must be distinguished from the consequences of putting it into effect.
7. Depending on the particular organization of the group of potential voters as well as depending on its objectives, casting a vote may range from being an instance of making a choice to being an instance of taking a decision, with in between acts of voting that constitute a class of their own. Here are two extremes:

- If the voting outcomes cannot be traced back to individual voters (anonymity), and if the voter cannot predict the impact of his/her particular choice made when voting, the IVO will not have the form of impact the voter expects from a decision outcome, and for that reason its coming about will not count as a decision.
- If the vote is open and if the voter can reasonably predict the impact of its own contribution to the aggregate outcome, the individual voting outcome may be considered a decision outcome and the preceding act of voting may be considered a decision.

8. In some cases a decision outcome needs to be confirmed (ratified) afterwards by some body with higher authority. Then the action producing that "decision outcome" still counts as a decision if it was taken with the expectation that ratification will succeed. This leads to the following pipeline:

- A process involving different levels and stages of decision taking and social choice leads to the listing of persons that may be elected in office.

³This form of concurrency can be adequately formalized with arbitrary interleaving based parallelism, for instance using the process algebras specified in [4].

⁴Making an aggregate social choice (and thus producing an AVO) as implemented by means of voting is in most cases not an instance of decision taking as understood in [6] because the aggregate agent has no intentions which it is aiming for and because it makes no predictions about the consequences of putting the AVO into effect. Social choice, is not an instance of choice and social choice is not necessarily an instance of decision either. By default it is not. In fact, In many political systems social choice is a more powerful mechanism than decision taking.

- Social choice mechanisms, often based on voting, are used to select who will be in office for some period.
 - Persons in office enact decision taking, for that task they may be supported by non-elected staff for decision making (decision preparation).
 - Decisions must be ratified afterwards by means of small scale social choice (normally involving non-anonymous voting), performed by bodies of agents who have been put in place for limited periods of time by social choice mechanisms implemented by way of anonymous voting.
 - The primary responsibility for the (consequences of effectuating the outcome of the) decision remain with the agent who took the decision, who is by default assumed to expect ratification.
9. Confidential boardroom voting with individual voting outcomes open to all participants may be in some cases be considered an instance of decision taking, whereas the voting that takes place with national elections in so-called democracies does not. Indeed I will assume that a small group of persons constituting a board, which is engaged in open and personal discussions, can have intentions in spite of possible differences of opinion, whereas a large group of individuals cannot be attributed an intention on the basis of the interpretation of an AVO that came out of a group voting process in which these individuals had been invited to participate.⁵

One may contemplate voting as a service (VaaS). In some cases it is admissible that an individual outtasks a voting assignment to another agent. It is implausible, however, that an agent specializes in voting on behalf of other agents on a systematic scale. For this reason I consider VaaS to be an implausible notion. Precisely this observation necessitates making a distinction between voting and deciding in preparation of contemplating DTaaS.

2.3 Promise issuing

Promising, that is, issuing a promise, is an action comparable to but yet different from taking a decision. This matter needs to be analyzed at this point because it impacts on the plausibility of DTaaS. For the notion of a promise I refer to [17, 18, 7]. As discussed in [6] there is a connection between promise and decision. This connection would be simpler and more symmetric if a promise were understood as an act of promising. Because that convention would deviate too much from common usage I will not primarily understand a promise as an act of promising but rather as its outcome. A promise is the tangible or intangible outcome of an act of promising and promise issuing (PI) is performing the act of promising.

⁵The social choice processes supposed to be supported by the forms of reasoning as suggested in [5] may also count as decision taking in my opinion.

There is remarkable complementarity between promise issuing and voting. Both voting and promise issuing produce an outcome from which further consequences may result. Voting involves choice, but it need not be the expression of an intention or of an expectation. Promise issuing involves intentions and expectations but it need not involve choice.

I will distinguish promise issuing from promise making, with promise issuing referring to the action, and promise making referring to a more comprehensive process of which promise issuing constitutes the concluding phase. Promise making may include preparatory steps, mainly for designing the form and the content of the promise (also called promise body in [7]).

Promise issuing is close to decision taking but it differs in important ways. I suggest the following relation: every decision outcome is a promise but not the other way around.⁶ Promise is more general than decision outcome because: (i) the promiser (comparable to the decision taker) need not have any expectations concerning the consequences of effectuation of the promise, (ii) the promise may be intangible (like a mathematical entity, which stands for a shared cognition) while a decision outcome must be tangible, (iii) the role of a promiser is immaterial, (iv) if a promise has an intangible form, for instance two or more agents remembering that something was said, then the consequences of the promise are mediated via the different components of the promise (that is the cognitive residues of the act of promising as they exist in different agents that were in scope of the promise, including the promiser) and not via the single “promise” (viewed as an outcome of promise taking), (v) the scope of a decision outcome is merely its readership as prescribed by the decision taking protocol at hand, whereas the scope of a promise (see [7]) is more immediately determined by the act of promising involved.

The reason to consider promise issuing in some detail arises from the following observation: “promise issuing as a service (PIaaS)” is a problematic (that is, not necessarily plausible) concept because (i) it is unclear how promise issuing can be delegated to another agent, (ii) it is unclear what it might mean to be more capable of promising on behalf of a promiser than the promiser is him/herself.⁷ I conclude that although DT is a subcategory of PI, for PI in general PIaaS is a problematic notion. At the same I intend to maintain that DTaaS is a coherent (potentially plausible) concept.

So it must be the case that especially for cases of PI that do not qualify as DT provision as a service is implausible. A criterion that separates promises for which PIaaS is implausible from decisions is the concurrent creation of obligations for the promiser. Some promises create obligations for the promiser.⁸

⁶The viewpoint that each decision is an instance of promise issuing derives from the formal definition of a promise in [7]. For different definitions of promise issuing the relation with decision may work out differently.

⁷“Marketing as a service (MaaS)” is a plausible notion, and some forms of MaaS might be viewed as an instance of PIaaS. The difficulty with understanding PIaaS is immediately connected to the difficulty of explaining the precise role of promises. The abundance of promises “in the real world” seems not to be based on the existence of a definite semantics of the term promise but rather on its absence.

⁸According to [17, 18, 7] promise issuing need not involve the creation of an obligation

Obligation creating promise issuing is not an instance of decision taking. It appears that “obligation creating promise issuing as a service (OCPIaaS)” is an implausible notion. That implausibility renders PIaaS problematic, but not DTaaS.

3 DT as an (organizational) feature (DTaaS)

Besides interpreting decision as an act rather than as an outcome I will consider it to be a feature of organizational structure, amenable for analysis in terms organizational design and architecture, rather than as an aspect of human behavior primarily amenable to empirical investigation. DTaaS is a perspective on DT which allows to assess from an external perspective which activities count as DT and which do not. DTaaS also allows the view that an organization may be transformed as to feature more (or less) DT, that is to have more (or less) frequent occurrences of activity classified as DT.

Decision taking is a concept that emerges from attempts to modularize the behavior of existing or imagined organizations, rather than a concept which emerges when considering the behavior of single individuals or the collective behavior of groups of individuals.⁹ I will follow [6] with the following assumptions:

1. A decision is an act of decision taking with actor (agent), time and place as required coordinates.
2. A decision must produce a decision outcome which is a tangible item for instance a text.
3. The effectuation of a decision outcome leads to the consequences of that decision outcome (sometimes abbreviated to “consequences of the decision”).
4. Every decision is taken by an agent with the intention that effectuation of the resulting decision outcome has the expected consequences; it depends on the role of the decision taker who needs to take care of effectuating the decision outcome.
5. If agent a takes a decision that is action is performed in the context of a playing some specific role, without that role being known or specified a ’s

for the promiser. In particular promise issuing by automated agents must be viewed as autonomous action.

⁹In psychological research it seems to be taken for granted that many forms of behavior, including most forms of human choice, may be considered instances of decision making, which is usually not distinguished from decision taking. I disagree with that usage of the language to the extent that in my view when making a choice is observed no decision taking is necessarily implied. Choice takes place whenever an agent acts in a setting where the agent was aware of alternative actions that it might have performed instead in full compliance with operational constraints of the setting. An animal may also perform choices even if it is harder to determine the meaning of awareness in that case. That an animal is able to take decisions is implausible according to [6].

production of a decision outcome amounts to no more than a mere speech act,

6. Rather than the choice between different options, the production of an intermediate product (the decision outcome) is the primary feature of a decision.¹⁰
7. In the absence of a decision outcome, the effectuation of which brings about the intended consequences (as expected by the decision taker), making a choice between different behavioral options amounts to a choice and a choice may be made in the absence of a decision.
8. The only obvious alternative to a decision is not to take that decision (with the same coordinates and producing the same outcome), (ix) decision taking is a concluding subprocess of decision making, decision making primarily involves for instance the preparation of decision outcomes; decision making is embedded in a larger decision making process which also involves protocol actions that do not influence decision outcomes.
9. A decision terminates an episode of indecision.
10. A decision is an action but it need not be a decisive action, while a decisive action need not be or involve a decision. A decision is a decisive action only if in hindsight it appears that effectuation of the decision outcome had decisive consequences.
11. Making up one's mind in preparation of an action is not an instance of decision taking.
12. A command is the outcome of an act of commanding (command issuing) and it is like a decision but it indicates in addition who who has to act. If the command is tangible it comprises a decision, but if it is intangible the command is a speech act which fails to count as a decision. Thus: command issuing that leads to a tangible command (outcome oriented command issuing) produces a decision at the same time, whereas on the fly command issuing produces an intangible command only which does not count as a decision.

In addition to what was stated in [6] it is required that a decision outcome is largely self-explanatory. A mere bit (0 or 1, usually termed “yes” or “no”) is insufficient in the absence of an unambiguous reference to a rendering of the question to which that bit is supposed to constitute an answer.¹¹

¹⁰Typically a car driver makes choices which do not qualify as decisions when handling the controls of the car while in motion. However, the step to install a car kit for a mobile phone, or the step to buy a new map or a (new) navigation device, may or may not be the consequence of a decision. For instance if a form is created and signed to instruct a car dealer to install a new car kit, that form may be understood as a decision outcome of the car owner's decision to acquire a new car device. But if besides buying food, fuel, and newspaper, a map is bought because of its fresh look and feel, the latter action may not count as having been done as an effectuation of a decision outcome.

¹¹When voting an agent may choose “yes” (or “no”) unaware of its implications or meaning.

3.1 Remote method invocation as an analogue

My perspective on decision taking is derived from a perspective on activity found in the area of computer programming. When observing a machine effectuating an instruction sequence one sees no more than a sequence of steps. That the instruction sequence has been written in a so-called structured notation is a matter of conceptual organization at a higher level of abstraction.

For instance it might be stated that a certain sequence of steps is best generated by effectuating a method call from an appropriate class thus providing the additional judgement that methods and classes are useful means of structuring in the case at hand. Seen from the perspective of an effectuating machine it may be preferable that a particular method is performed by another machine, which comes close to it being outtasked by that machine, thus leading to remote method invocation, or in terms of our topic: method effectuation as a service (MEaaS).

Rather than viewing decisions as very particular actions taken by agents or by organizations, decision may be seen as a way to structure organizational behavior. For instance it may be “decided” (at the level of organizational design) that certain actions need to be preceded by decisions so that these actions can be understood as the effectuation of decision outcomes. That is useful only if a protocol for decision taking, as well as for the preparatory part of decision making is laid down as well.

From this perspective it is plausible that if an organization U plans to make use of DTaaS it may need a preparatory phase for structuring its operations in such a way that DT takes a more prominent place to begin with. In particular DT may then take place as a thread amongst other threads in a multi-thread run by way of strategic interleaving by some managing body.¹² The transition towards DTaaS where some external agent s provides a similar thread as a service to U ’s management body may be understood as temporary thread migration.

DT as provided by an internal service must be monitored internally from the perspective of DT demand management. Only if it is known what non-DT threads expect from a DT thread it can be agreed what to expect from an external provider of a DT thread.

3.2 Decision-free operation

The presence of a systematic DT process is likely to lead to bureaucratic overhead. Introduction of DTaaS seems to work in the opposite direction. Application of DTaaS is only possible if DT can be seen as a disposable feature. An organizations existence need not involve much decision taking. Here is a list of what

In our understanding of the act of voting a vote may be cast by an agent without having any intentions in mind.

¹²Changing an organization into a mode of operation where decision taking is more prominent has virtues outside the context of DTaaS. Achieving such transformations may be valuable if increased accountability and transparency are sought. Of course the converse holds as well: turning decisions into mere choices may be helpful if accountability, traceability, or transparency are to be diminished.

can be performed outside decision taking.

1. Can an organization can do without DT altogether, and if so to what extent? Yes, and indeed many activities and actions while not counting as DT lead to results comparable to DT:
 - meeting, conferencing, brainstorming,
 - monitoring, logging, writing and distributing meeting minutes
 - data mining, desk research, interviews, opinion polling,
 - creation and mining of feelings and opinions, possible mediated by interaction promoting IT tools,
 - planning,
 - promising (for promises not qualifying as decision outcomes, see Section 2.3 below.
 - voting, social choice,
 - choice from different alternatives, followed by action,
 - choice from different alternatives followed by plan formation (though without taking action),
 - improvised action,
 - deliberate individual action,
2. Is top management intrinsically connected with DT? No, management tasks may be confined to:
 - providing a sense of direction, including a mission
 - providing objectives and goals including a strategy,
 - providing a policy preferences and tactical options.

Each of these aspects can be influenced by means of informal steps which fail to qualify as DT.

3. Mission statements, strategic directions, and policy preferences can be set without making use of DT. Of these matters have been communicated effectively to the workforce of U each of its members is in a better position to act when actions involve making choices that may discard options, or to perform actions that will create new options for future activities of U .

4 On services and service casting

A service is defined as an identifiable and intangible activity that is the main object of a transaction in [39] (see also [20]). I will use a somewhat more detailed definition of a service, useful in particular to separate services from products, that was used in [30]. In [30] the distinction between products and services is

clarified by means of an extensive quote from [44]. In brief a service is sold by one party to another, but as opposed to product, it is intangible (as an entity), heterogeneous (as a concept), it is perishable (non-enduring), and it is necessarily consumed and produced concurrently. According to [30], however, services may carry products along and products may carry services along, the difference being a matter of gradation rather than a very sharp one. One may perhaps simplify these requirements to the condition that a service is a process which can be provided against compensation by a third party.

4.1 Casting activities and products as a service, a catalogue of examples

In [23] “music as a service (MaaS)” is explained to be the transfer of music in digital form, by way of streaming without transferral of ownership of the data. MaaS is seen as an instance of content as a service (CaaS). Now if one listens to ordinary live music the suggestion that ownership is transferred has never been present, and that suggests that live music has always been music as a service. However, in order to grasp the phrase “music as a service” one needs to understand that live music is not meant as it would be odd to look for a new name for such a classical phenomenon. For that reason the phrase MaaS lives in another world, most plausibly the digital world, where music as a product is known, and music as a service may be a novel phenomenon. More can be said, however: the provider of live music cannot concurrently serve different customers with different tunes, whereas a digital service provider is supposed to be able to serve different clients at the same time in ways more flexible than mere broadcasting. This concurrency criterion on service provision is absent in the discussion of [30] but it seems to be a helpful addition to understanding what is amenable to service provision and what is not.

In [33] business dashboards as a service are discussed. As far as I can see the authors fail to explain the plausibility of service casting in this case. Instead they merely analyze the virtues of dashboards (implemented as an interactive tool for business data visualization) for management purposes as well as the need for further research about them. The phrase “dashboards as a service” suggests the presence of a third party collecting a client’s business information and extracting from that a clever abstraction which is provided to the client in the form of a dashboard, which itself is an internet service. This is a plausible explanation of what “dashboards as a service” might amount to, irrespective of the extent to which that constitutes a realistic IT market opportunity at this moment in time. In the discussion of [30] from which we have already quoted criteria for the distinction between product and service, the presence of a third party arises from their condition that a service can be sold by one party to another.

In [26] the usage of “(security) policy as a service” is explained in the context of cloud security. This is a convincing example of service casting. A security policy is intangible and non-product like but its real time provision by a third party (and hosted in the cloud) may be understood as a novel feature.

In [35] one finds an elaborate description of (software) “composition as a service” (abbreviated as CaaS in [35] but here as SCaaS in order to avoid confusion with “content as a service”). SCaaS is presented as a phenomenon in the context of software as a service (SaaS). SCaaS differs from SC by its provision by a third party which justifies service casting in this case. Software composition satisfies all requirements in the entities traded as services as formulated in [30].

In [16] a description of cloud computing is given with a focus on “infrastructure as a service” (IaaS). IaaS is considered the most generic service provided by a cloud in addition to PaaS (platform as a service, see also [27]) which is committed to some fixed systems software, and SaaS (software as a service) which in addition implies a commitment to an application area and a family of software products applicable to that area from which services may be composed automatically on demand. The phrase IaaS is difficult to grasp because infrastructure primarily refers to a product rather than to a process. It seems to be more precise to speak of “infrastructure behavior as a service” (IBaaS) instead.

In [1] flexibility as a service (FaaS) is proposed. Although flexibility is no more than an attribute of a service offering, this instance of service casting is based upon the fact that principles of service composition can be provided in such a way that flexibility results. Remarkably, “services as a service” (SVaaS) becomes meaningful if it implies a focus on service construction by means of systematic composition in real time at the provider side. As [1] points out, FaaS is an objective which can be obtained through what I just termed the SVaaS perspective.

Software as service is currently the most well-known instance of service casting, by which all other instances seem to have been inspired.¹³ I will now discuss this example of service casting in some detail.

Finally an interesting case is “politics as a service (POLaaS)”.¹⁴ POLaaS (often written PaaS, which unfortunately collides with “platform as a service”) has a formidable presence on the internet.

4.1.1 SaaS, SEaaS, SENaaS, STaaS, and SPaaS

In spite of software constituting a product (tangible good) rather than a process (intangible good), SaaS (software as a service, see [40]) is at present the most prominent instance of service casting.

Because software is effectuated when used, “software effectuation as a service” (SEaaS) makes perfect sense, (with AHaaS for “application hosting as a

¹³If software is identified with “families of polyadic instruction sequences”, an identification which I consider to be adequate, it becomes reasonable to understand software as a mathematical entity, which is intangible rather than tangible. Of course the physical representations of mathematical objects are tangible, but the mathematical objects themselves are not, or not necessarily. Thus I will identify computer software with “(conventional) physical representations of families of polyadic instruction sequences”, thereby rendering software tangible.

¹⁴The late Dutch politician Pim Fortuyn made headlines in the Netherlands with his slogan “at your service”, which may be understood as his intention to provide POLaaS to the Dutch public. The extent to which POLaaS is a credible option that escapes from the problems of plain power oriented populism still puzzles Dutch political commentators ten years after Fortuyn’s violent death in 2002.

service” as the more common name). SaaS goes beyond SEaaS however, as it also comprises the on demand and real time composition of services (as provided by software being effectuated), and it may include the provision of software from a remote server. SEaaS and SENaaS are both included in SPaaS (software process as a service), which goes beyond SEaaS and SENaaS by taking all phases of the software life-cycle into account. SPaaS also comprises the service area STaaS (software testing as a service, also known as TaaS) and SDaaS (software development as a service). Another option is PRSaaS for providing software as a service.

4.1.2 (Provision of software) as a service = provision of (software as a service)?

The background of all service casting expressions XaaS might, if only as a thought experiment, be understood as follows: (i) XaaS is a service, (ii) every service H is identified with providing H, (which is abbreviated to PR H). Thus: $XaaS = PR(XaaS)$, (iii) associativity of bracketing is used $PR(XaaS) = (PR X)aaS$. In the case of software:

Software as a service = providing (software as a service) = (providing software) as a service.

Identification of a service casting with its provision may be considered imprecise, and that leads to contemplating inverse-provision as a kind of inverse operator for service casting. If inverse-provision denotes the abstraction from the phenomenon of the provision of a service to the abstraction consisting of its underlying service then one obtains the equation:

$XaaS = \text{inverse-provision}((\text{provision of } X) \text{ as a service}).$

This explanation of service casting would have the effect that it renders every service casting plausible, which is itself an implausible state of affairs, which for that reason I will not take it seriously.

4.2 Plausibility of service casting

I will use concept as a category that includes, entity, service, process, product, and method. For each concept X, the service casting of X, [X]aaS can be contemplated. Service castings range from implausible to plausible. X satisfies the so-called service casting preconditions of Paragraph 4.3 below if and only if [X]aaS is a plausible service casting.

The question arises which cases of service casting are plausible. That question is seemingly paradoxical in the following sense: if X is a service then XaaS makes little sense because [X]aaS suggests that X is seen from the service perspective, instead of its “natural” perspective.

4.2.1 Implausible cases of service casting

If X is not nearly a service then it may defeat being viewed from a service perspective. Here are some examples. None of the following service castings seems

to make sense: “jogging as a service”, “meaning as a service”, “gold ownership as a service”, “medication as a service”, “bicycles as a service”, “hardware as a service”, “sleeping as a service”, “breathing as a service”, “flying as a service”, “consciousness as a service”, “being born as a service”, “self-confidence as a service”, and “dining as a service”.

4.2.2 Unclear cases of service casting

There are unclear cases as well. At this moment I have no judgement available on the plausibility of the following potential service castings: “choice as a service”, “social choice as a service”, “owning as a service”, “spying as a service”, “swimming as a service”, “natural numbers as a service”, “walking as a service”, or “happiness as a service”.¹⁵

4.2.3 Service casting obstacles

Service casting obstacles are preconditions on a concept X that render XaaS implausible under all circumstances. Two such conditions can be identified at this stage: X is a service already (a violation of this condition will be labeled as weak implausibility of the service casting at hand), and non-disposability, that is after outtasking or outsourcing X from *U*, an unacceptable loss of identity of *U* has occurred.

4.3 Service casting preconditions

The following conditions on X must be met for [X]aaS to be plausible:¹⁶

Ontological constraints. Preferably X satisfies the criteria listed in [30] for services. To meet those criteria X needs to be intangible, and non-enduring, and P’s production and consumption must be necessarily concurrent. Preferably X is a kind of activity rather than a kind of entity. If X is tangible the following clause applies.

Service casting preconditions for product classes. If X is product-like (tangible) rather than service-like (and therefore intangible) and if [-]aaS is instantiated to XaaS, then the following conditions must be met:

1. XaaS will provide processes P that usually represent the usage of X (or of instances of X),¹⁷ and,

¹⁵In contrast “well-being as a service” (often referred to as wellness service) is plausible.

¹⁶Plausibility is a semantic notion purely related to the amenability of X for being offered as a service, or for suggesting a clear interpretation of [X]aaS which is to be distinguished from usefulness of [X]aaS, once it has been accepted as a plausible concept, for its consumer and producer.

¹⁷If XB represents the behavior of an X (or of an instance of X) then XaaS is understood as XBaaS by default.

2. if a process P representing the usage of X (or of an X) can be provided through the internet, services as meant by HaaS will be delivered as a webservice, and several customers may be provided the same service at the same time.
3. if P is most likely is to be provided via the internet, and if normal effectuation of the process X does not involve human operation then HaaS will definitely not involve human operation.¹⁸

Significant deviation from normal casting. The concept X is not normally seen as a service.¹⁹ A violation of this precondition amounts to weak implausibility. Weak implausibility does not imply incoherence, it merely implies service casting is applied in an implausible case. From these considerations it follows that casting activity, or entity, X as a service requires that the audience for the phrase “X as a service (abbreviated XaaS)” accepts the following implicit ambiguity:

- X must not a service by default. But the difference between the default role of X and its role after being cast as a service should not be too large either. If it is too large one arrives at implausible service castings, some of which have just been listed.
- The more distant XaaS is from the normal role of X the more successful the notion XaaS may become. Probably SaaS has acquired significant popularity just because software is commonly understood as a product.

self X is the normal (default) case. If X is normally a service it can be placed in the context self[-] thus obtaining “self[X]”.²⁰ If it is explicitly meant that X is understood in its default meaning that can be expressed with “X as usual”.²¹

Demand management phase is an option. As a preparation for a XaaS phase for its prospective consumer *U* it may need to structure its own processes in such a way that X is performed as self[X] by *U*, though in a separate unit which is monitored and managed with principles of demand management.

Disposability. It must be the case that stripping X from a unit need not necessarily imply a loss of its identity.

Concurrency potential. Implicit in *a* offering a service X to *U* is that *a* can, at least in principle, offer the service X concurrently to a plurality of clients.

¹⁸In the case of SaaS (software as a service) this constraint explains why SaaS will not extend to “software engineering as a service” (SENaaS), a “product” involving human engineers that has been delivered by the software industry for many years.

¹⁹I propose to call “online banking as a service” weakly implausible because online banking is normally considered a service.

²⁰For instance self hairdressing.

²¹For instance: “self hairdressing” is probably cheaper than “hairdressing as usual”, thus avoiding “hairdressing as a service” which is weakly implausible.

This form of concurrency can be described by strategic interleaving for multi-threading as outlined in [12]. That concurrent offering of X is an option is a property of X and it can be understood as a service casting precondition.

Potential for co-creation of the service by provider and consumer. It is sometimes proposed as a key aspect of services that these are co-created by producer and consumer.

5 DTaaS

After considerable preparations a “formal” (in the sense of “official”) introduction of DTaaS is now enabled. The introduction splits in four parts: (i) creating DTaaS as a (meaningful) concept, (ii) a survey of DT specific aspects of DTaaS instances, (iii) a survey of potential advantages of DTaaS for its consumer and provider, and (iv) a survey of risks and potential disadvantages for RTaaS.

5.1 DTaaS: a definition

As a concept, DTaaS is the result of service casting applied to the concept DT. For this result to be well-defined compliance with the various service casting preconditions needs to be checked. This matter will be addressed in Paragraph 5.1.1 below.

Each instance of DTaaS involves a unit and an agent. DTaaS refers to a particular kind of sourcement: DTaaS occurs if an agent a provides DT as a service to a unit U . In the case that a provides DTaaS to U , the decisions taken by a are taken on behalf of U and have the same status as decisions taken by U ’s management.

Much can be said about instances of DTaaS which is specific for DT as a concept. For instance that this sourcement is likely to have arisen by outtasking a part of U ’s DT to a , where outtasking refers to a temporary transfer of an activity (task) to another unit (the unit containing a) without a corresponding transfer of sources. In Paragraph 5.2 a range of further observations concerning instances of service casting DT, which are specific to DT rather than to service casting in general, is made.

5.1.1 DT meets service casting preconditions

DT is intangible, and it is not normally cast as a service. Further self DT is the default situation and demand management for DT is an option, though in order to understand that option one probably needs an awareness of the notion of DTaaS already. DT is disposable in some cases and a DTaaS provider may concurrently provide DT services for different clients. By developing adequate DT protocols, a DP task which may be shared with the client, co-creation of DT services between provider and consumer may be achieved.

5.1.2 DTaaS avoids service casting obstacles

DT is not usually considered a service, and outtasking DT need not degrade a unit's identity. In 3.2 a listing has been made of the activities that differ from DT and that will remain after outtasking DT.

5.2 DTaaS: DT specific aspects

The purpose of this section is to specify the result of service casting of DT, and more specifically of DT_I (leading to DT_{IaaS}) for some decision interface I , in greater detail than merely stating that it results from service casting.

5.2.1 Which decisions can be delegated?

Which decisions can be delegated to a service provider? Because it is hardly conceivable to hand over “all DT” to an external agent, there is no way around some form of modularization. To that end one may take advantage of the concept of a decision interface (DI), which collects a coherent family of issues or topics about which decisions may be taken. Elements of a decision interface are sometimes called decision rights, but that is asymmetric as these elements may equally well be understood as decision obligations (or more neutral, decision tasks, decision options, or decision patterns). The notion of a decision interface is symmetric. It is understood that an agent having regular DT as its task may collect all of its potential decisions in an overall DI which is modularized (decomposed) as a combination of disjoint sub DI's.

Given a decision interface I , DT_I represents DT concerning decisions in I , DM_I stands for DM concerning decisions in I and DP_I concerns DP limited to decisions in I . We have the following I -specialized equation “ $DM_I = DP_I + DT_I$.”

In [6] it has been argued that DT requires that the the DT agent operates in a clearly determined role. Thus some role must be provided, at the least this role is the following “external DT agent on behalf of U ”, or “external DTaaS provider for U ”, or simply “external decision taker for U ”. The role name is likely to be linked to the decision interface for which the role has been sought.

It is an existing practice that external DTaaS is sought for brief episodes and dedicated to a single issue or to a very confined range of issues, the role of an external DTaaS provider has more intrinsic names, for instance a jury, if a prize is to be awarded on behalf of U ,

Further examples of role names for thematic episode driven instances of DTaaS (abbreviated as EdDTaaS) with limited scope: (i) a mediator if mediation is sought for a conflict between subunits within U (and DTaaS is applied because m_U may not be considered sufficiently impartial, or may not have the required authority or trust base within U , (ii) a court if U has sought the intervention of a legal process, (iii) some specialized authority (e.g. the EFSA, European Food Safety Authority) which is asked to intervene, (iv) an active private banker who may trade on behalf of its clients, (v) a mountaineering guide

is in charge of a well understood part of his/her client's DT during a trip, (vi) an auctioneer decides about transactions between parties who have each agreed to stay within the protocol prescribed by the auctioneer.

5.2.2 What sourcing options exist for DT?

Having stated that an instance of DTaaS is a sourcing of a particular kind, it is useful to consider in some detail which sourcements may allow an external agent a to perform DT (or DT_I) on behalf of the management m_U of some unit U ? Here are some observations concerning that matter.

- Unit management may be self-sourcing for DT, that is m_U performs DT (in other words it applies self DT). This is most common and I will assume that it is the case by default.
- DT may have been delegated by m_U to one or more subunits at a lower level in U 's management hierarchy. In this case U is self-sourcing for DT but m_U is not.
- U may well distinguish between governance (g_U) and management (m_U), in which case it is the role of g_U to delegate DT to management functions within U .
- if g_U temporarily delegates DP to an interim manager im_U instead of to m_U , U is still self-sourcing for DT because im_U operates under the responsibility of g_U .
- U may be temporarily non self-sourcing for DT. That means that DT has been temporarily delegated (outtasked) by g_U to an agent a operating outside U . In this case a acts as a service provider for m_U . The authority and steady support of g_U is required to assure that a is in a decision taking role rather than merely in a consulting role where it may at best suggest potential decision outcomes to m_U .

As long as g_U keeps the outtasking relationship between U and a in place m_U accepts the decision outcomes of a decisions as if these were decision outcomes brought about by its own decision taking. It is essential to specify in advance which types of decisions constitute part of the DTaaS agreement, because m_U cannot undo a 's decisions any easier than its own decisions.

5.2.3 Which sourcing transformations exist for DT?

In [11] the notion of a sourcing transformation has been examined in detail, with a particular focus on the following transformations: outsourcing, insourcing, back sourcing, and follow-up outsourcing. Without further analysis I will postulate that corresponding transformations can be found if only tasks but no sources are being transferred: outtasking, intasking, backtasking and follow-up

outtasking. For the case of DTaaS the outtasking transformation is of particular importance.

- DT cannot be outsourced. This is so because outsourcing of DT (from a state where it is performed by m_U) involves moving m_U (or a part of it) outside of U , which cannot be done unless the characteristic feature of its being the management of U is lost. An artificial option which I will discard as being unconvincing is to give members of m_U part-time positions in an insourcing unit U' outside U .

Another artificial option that is mentioned only is that given the decision interface I , the body m_U is decomposed as $m_U = m_U^{d,I} \cup m_U^{nd,I}$ into disjoint groups $m_U^{d,I}$ (involved in decision taking about decision patterns in I) and $m_U^{nd,I}$ (not involved in decision taking concerning decision patterns in I), and to assume that $m_U^{nd,I}$ is outsourced by U to a . This option is artificial because it casts doubts on the status of m_U as a managing body in the original situation.

- outtasking of DT is possible, a consequence that being that the tasks of m_U are temporarily reduced.
- Task externalization is not an option for DT. Indeed it is against the required autonomy of U as being an independent unit if DT is transferred to an external agent on an indefinite basis. Thus the feature of outtasking that it comprises a temporary transfer of tasks is essential in the case of transferral of DT.
- If DT has been outtasked to agent a , that agent is called an external decision taker on behalf of U .
- Because DT cannot be outsourced and DT is a part of DM, only partial outsourcing of DM is possible. In particular only partial or full outsourcing of PM is possible.
- Typically DP or parts of it can be outtasked to a so-called consultant.
- By definition DT cannot be outtasked to a consultant (in its role as a consultant).

5.3 Possible reasons for using DTaaS

DTaaS is a (potentially) meaningful concept in the sense that it may open a novel perspective on organizational architecture. DTaaS seems to be uncommon except in cases where quite dedicated DT processes are outtasked and often in a case by case fashion only.

DTaaS will be initiated by the outtasking unit U , or rather by its governing body (subunit) g_U which will make use of the DT services of agent a which may be a part of unit U' instead of having that part of DT self-sourced for U by m_U .

For doing that g_U needs plausible reasons. These reasons will differ from case to case but some general remarks concerning these reasons can be collected:

- DT provider a may be less vulnerable to the effects of stress created by an involvement in DT concerning U than m_U . Even a 's authority or trust base may exceed that of m_U .
- DT provider a 's awareness of the consequences of possible decisions to parties outside U may significantly exceed that of U .
- DT provider a may include highly skilled DT specialists who are especially competent in: (i) timing of DT, (ii) prediction of the effects of implementing potential decision outcomes, either tactically or strategically, or both, (iii) managing the core of DP preceding DT, (iv) evaluating the quality of preparatory DP activities may exceed that of m_U , (v) drafting decision outcomes, (vi) communicating decision outcomes.
- DT provider a may be able to apply process models to DT which it has used elsewhere and by doing so perform at a higher level of competence compared to the competence level provided by the DT function within U . For instance a may be well placed to deal with (that is to guarantee) abstraction (preventing information from leaking to third parties) and encapsulation (preventing external interference) for the decision process.²²
- Every decision may be understood as the effectuation of a decision taking thread, itself the result of putting into effect an underlying instruction sequence (e.g. see [14]). Effectuating such instruction sequences²³ may be a particular competence of a .
- As argued in [6] DT may involve multi-threading (see [12]) in the case of DT for U . As a may be providing DTaaS services for several different clients simultaneously a may be dealing with hierarchical multi-threading (see [13]). More efficient methods for strategic interleaving of hierarchical decision threads may be available to a than to m_U .
- By outtasking DT m_U arrives at a situation where it can concentrate on putting decision outcomes into effect. In that case m_U may operate in an almost decision free fashion. Once decision outcome effectuation has become standard practice m_U may terminate outtasking DT or part of it.
- For U it may be helpful to forget about short term tactical DT in order to fully concentrate on long term strategic matters. The opposite may hold as well: m_U makes use of DTaaS provided by a hoping that a is able to deal with long term and strategic matters while m_U is capable of operating with full concentration on tactical and sort term issues of vital importance to u 's existence.

²²Abstraction and encapsulation are notions from process theory ([4]) which find an application in many decision making process.

²³In particular these instruction sequences may involve conditions written in the notation of short-circuit logic, which is semantically analyzed by proposition algebra (see [15]).

- DTaaS can be used to reduce the bureaucracy in the outtasking organization. Bureaucratic simplification can conceivably be achieved by training staff to perform a significant amount of action without the need for DT processes, thus reducing the volume of DT and outtasking a part of the remaining DT may further reduce the dependency on DT. Once DT is out of the way business process optimization and automaton can both be used to streamline an organization's workflow.
- DT provider a shares the responsibility of its decisions, and concerning the consequences with the unit U to which the service is provided. The legitimacy of a having these responsibilities has several sides, but a sufficient criterion is that g_U which is in charge of ultimate responsibilities for U can argue convincingly that it is in U 's best interest to outtask DT to a . In particular if an emergency of a particular kind has to be dealt with, a 's competences to deal with the variety of issues connected with that particular kind of emergency may be valued of higher importance than conformity of a 's own intentions with achieving some or all of U 's strategic objectives.

5.4 DTaaS risk analysis

There is a plethora of potential risks that DTaaS carries with it for its consumer. I will mention only a few aspects of the risk analysis.

- There may be a risk of vendor lock in when U makes use of DTaaS provided by a . If DP is outsourced to a there is a risk that U may not be able to realize back sourcing of its DM activities, thus becoming dependent of a . This is a reason for not outsourcing DP to the same service provider as the one to which DT is being outtasked. It even suggests that outsourcing DP brings with it a higher risk of vendor lock in than outtasking DT may do. If only DT has been outtasked to a , U must make sure, and communicate that backtasking DT from a is an option at regular instances of time.
- Like in the case of an external consultant U 's appreciation of a 's services may degrade. In that case a 's staff may withdraw from its involvement in a DTaaS sourcement if its authority has become problematic and its activity is less effective.
- Is DTaaS a common practice or is it a mere thought experiment? DTaaS for short episodes or limited to relatively confined decision classes is common practice, but DTaaS for the full width of m_U decision rights seems to be unusual. That creates the risk that an occurrence of DTaaS is taken to be a sign that m_U fails to live up to its expectations. In fact m_U seems to carry an obligation to take the majority of its own decisions itself. DTaaS is about the thought experiment that this obligation is replaced by a less restrictive policy towards the range of task descriptions suitable for a management agent or team m_U .

- Assuming that a provides DTaaS to different units a conflict of interest arises if at the same time a needs to strive for opposite objectives.²⁴ Because a is taking decisions it does so with the awareness of expected consequences (of effectuation of decision outcomes) and with the (delegated) intention to effect these consequences. That intentions can be delegated at all is open for discussion, and I will assume that when acting as a DT provider for U , a may pretend intentions (on behalf of a) that are not quite its own, provided that these intentions comply with a 's best and own intention to maintain a bundle of pretended intentions on behalf of U that best serves U 's interests, and to do so in a consistent fashion.

6 Concluding remarks

This paper contains a theory of DTaaS and following [10] it is fitting to ask about the intended audience of that theory and to ask what members of that intended audience might get out of it. In the terminology of [10] the latter question is phrased as to what ability or which conjectural ability is supposed to result from an acquaintance with this theory. These conjectural abilities are twofold: the ability to acquire a DT task and to start acting as a DTaaS provider (intasking DT, the transition complementary to outtasking DT), and the ability to outtask part of one's DT activities in order to concentrate on other activities which are considered to be of higher importance in some stage. Returning to the matter what constitutes an intended audience for this paper: those potential readers who take an interest in decision taking and have accepted the preparatory analysis made in [6].

An analysis of service casting has been provided as well as a demarcation of decision taking separating it from choice, individual voting, social choice, and obligation creating promise issuing. Using these ingredients the conceptual coherence (consistency) of DTaaS has been established and potential advantages of its use have been indicated.

Further work can be descriptive, aiming at finding instances of DTaaS in existence and investigating how that came about and what advantages are actually obtained. Another and perhaps more attractive path is to investigate existing organizations or projects and to find suggestions for first making DTaaS more visible in the organization (or project) and subsequently outtasking a part of DT so that DTaaS is made use of.

As an application of this work DTaaS can be introduced in pilot organizations or projects in order to find out which of the mentioned potential advantages can be achieved in practice, and if so under what circumstances.

²⁴Less apparent than an outright conflict of interest is so-called feature interaction (see [25]). If a is involved in providing DT services for different units it needs to analyze feature interaction between its respective decision interfaces. A conflict of intentions is an obvious instance of feature interaction, but more subtle interactions will need attention as well.

References

- [1] W.P.M. van der Aalst, M. Adams, A.H.M. ter Hofstede, M. Pesic, and H. Schonenberg. Flexibility as a service. *in: L. Chen et. al. proc. DASFAA 2009 Worksops, LNCS 5667*, pp. 319–333 (2009).
- [2] P.S. Atiyah. Promises, morals and law. *Clarendon Press, Oxford*. (2003).
- [3] S.R. Atkinson and A. Goodman. Influencing network decision taking. *Defence Academy of the United Kingdom, Adv. Research and Assessment Group Occasional Series*, no. 11 (2008)
- [4] J.C.M. Baeten, T. Basten and M.A. Reniers. Process Algebra: Equational Theories of Communicating Processes. *Cambridge Tracts in Theoretical Computer Science*. Vol. 50, (2009).
- [5] J.A. Bergstra. Real Islamic Logic. [arXiv:1103.4515 \[cs.LO\]](#), (2011).
- [6] J.A. Bergstra. Informatics Perspectives on Decision Taking. [arXiv:1112.5840 \[cs.OH\]](#), (2011).
- [7] J.A. Bergstra and M. Burgess. A static theory of promises. [arXiv:0810.3294 \[cs.MA\]](#), (2008).
- [8] J.A. Bergstra and M.E. Loots. Program algebra for sequential code. *Journal of Logic and Algebraic Programming*, 51 (2), pp.125–156, (2002).
- [9] J.A. Bergstra, G.P.A.J. Delen, and S.F.M. van Vlijmen. Introducing source-ments. [arXiv:1107.4684 \[cs.SE\]](#), (2011).
- [10] J.A. Bergstra, G.P.A.J. Delen, and S.F.M. van Vlijmen. Outsourcing competence. [arXiv:1109.6536 \[cs.OH\]](#), (2011).
- [11] J.A. Bergstra, G.P.A.J. Delen, and S.F.M. van Vlijmen. Stratified Outsourcing Theory. [arXiv:1110.1957 \[cs.SE\]](#), (2011).
- [12] J.A. Bergstra and C.A. Middelburg. Thread algebra for strategic interleaving. *Formal Aspects of Computing*, 19 (4) pp. 445–474, (2007).
- [13] J.A. Bergstra and C.A. Middelburg. Thread algebra with multi-level strategies. *Fundamenta Informaticae* 71(2/3), pp. 153–182 (2006).
- [14] J.A. Bergstra and C.A. Middelburg. Instruction sequence processing operators. *Acta Informatica* 49 (3), pp. 139–172 (2012).
- [15] J.A. Bergstra and A. Ponse. Proposition Algebra. *ACM Transactions on Computational Logic*, Vol. 12 (3) Article 31 (36 pages), (2011).
- [16] S. Bhardwaj, L. Jain, and S. Jain. Cloud computing: a study of infrastructure as a service (IaaS). *in: International Journal of Engineering and Information Technology*, Vol 2 (1) pp. 60–63, (2010).

- [17] M. Burgess. An approach to understanding policy based on autonomy and voluntary cooperation. in: *Ambient Networks, Springer LNCS*, Vol 3775 pp. 97–108, (2005).
- [18] M. Burgess. System administration and the scientific method. in: J.A. Bergstra and M. Burgess (editors), *Handbook of Network and System Administration*, pp. 689–728, (2007).
- [19] D.K. Cohen, J.G. March, and J.P. Olsen. A garbage can model of organizational choice. *Administrative Science Quarterly* Vol. 17 pp. 1–25, (1972).
- [20] E. Chung and C. McLarney. The classroom as a service encounter: suggestions for value creation. *Journal of Management Education*, Vol. 24. (4) pp. 484–500 (2000).
- [21] G.P.A.J. Delen. Decision and control factors for IT-sourcing. in: J. A. Bergstra and M. Burgess (editors), *Handbook of Network and System administration*: pp. 929–946, (2007).
- [22] A. Errington. The delegation of decisions on the farm. *Agricultural Systems*, Vol. 19. pp. 299–317 (1986).
- [23] J. Doerr, A. Benlian, J. Vetter, and T. Hess. Pricing of content services—an empirical investigation of music as a service. in *M.L. Nelson et. al. (Eds.): Sustainable e-Business Management*, LNBIP 58, pp. 13–24, (2010).
- [24] G.P. Huber and R.R. McDaniel. The decision-making paradigm of organizational design. *Management Science*, Vol. 32 (5), pp. 572–589, (1994).
- [25] K. Kimbler and L.G. Bouma (Eds.) Feature interactions in telecommunications and software systems V. IOS Press, (1998).
- [26] U. Lang and R. Schreiner. Analysis of recommended cloud security controls to validate OpenPMF “policy as a service”. *Information Security Technical Report*, 16 pp. 131–141 (2011).
- [27] G. Lawton. Developing software online with platform-as-a-service technology. *IEEE Computer*, Vol. 41 (6) pp. 13–15 (2008).
- [28] D. Mahalel, D. Zaidel, and T. Klein. Driver’s decision process on termination of the green light. *Accident Analysis and Prevention*, Vol. 17 (5) pp. 373–380 (1985).
- [29] C.A. Middelburg. Searching publications on software testing. [arXiv:1008.2647 \[cs.SE\]](https://arxiv.org/abs/1008.2647), (2010).
- [30] F. Niessink and H. van Vliet. Software maintenance from a service perspective. *Journal of Software Maintenance: Research and Practice*, Vol. 12. (2) pp. 103–120 (2000).

- [31] P.C. Nutt. Making decision-making research matter: some issues and remedies. *Management Research Review*, Vol. 34 (1) pp. 5–15 (2011).
- [32] M.J. O’Fallon and K.D. Butterfield. A review of the empirical ethical decision-making literature. *Journal of Business Ethics*, Vol. 59, pp. 375–413 (2005).
- [33] K. Pauwels, T. Ambler, B.C. Clark, P. LaPointe, D. Reibstein, B. Skiera, B. Wierenga, and T. Wiesel. Dashboards as a service: why, what, how and what research is needed? *Journal of Service Research*, Vol. 12. pp. 175–189, (2009).
- [34] W.H. Riker and P.C. Ordeshook. A theory of the calculus of voting. *The American Political Science Review*, Vol. 62 (1) pp. 25–42 (1968).
- [35] F. Rosenberg, P. Leitner, A. Michlmayr, P. Celikovic, and S. Dustdar. Towards composition as a service— a quality of service driven approach. in *Proc. IEEE Conference on Data Engineering*. DOI 10.1109/ICDE.2009.153, (2009).
- [36] M. Sengul, J. Gimeno, and J. Dial. Strategic delegation: a review, theoretical integration, and research agenda. *Journal of Management*, Vol. 38. (1) pp. 375–414 (2012).
- [37] A.G. Sheard and A.P. Kakabadse. A role-based perspective on leadership decision taking. *Journal of Management Development*, Vol. 26. (6) pp. 520–622 (2007).
- [38] H. Sheinman. Introduction: promises and agreements. in *Ed. H. Sheinman, Promises and Agreements, Oxford University Press*. pp. 3–57 (2011).
- [39] M.S. Sommers, J.G. Barnes, W.J. Stanton, M.J. Etzel, and B.J. Walker. Fundamentals of marketing. *McGraw Hill Ryerson, Toronto*, 7th edition (1995).
- [40] M. Turner, D. Budgen, and P. Brereton. Turning software into a service. *IEEE Computer*, Vol 36 (10) pp. 38–44 (2003).
- [41] M. Valdman. Outsourcing Self-Government. *Ethics*, Vol. 120. (4) pp. 761–790 (2010).
- [42] A. Whiteley and E. Wood. Decision making and decision taking: GSS and complex adaptive systems theory. *GBS Working papers, Curtin University of Technology*, No. 20 (2000).
- [43] H.P. Young. Condorcet’s theory of voting. *The American Political Science Review*, Vol. 82. (4) pp. 1231–1244 (1988).
- [44] V.A. Zeithaml and M.J. Bitner. Service Marketing. *McGraw Hill New York, NY* (1996).